White Paper
August 2014

IT@Intel
Evaluating Apache Hadoop* Software for Big Data ETL Functions

We determined that using Hadoop for ETL functions works well for datasets that are coming from, passing through, or resting in Hadoop.

Executive Overview

Intel IT recently evaluated Apache Hadoop* software for ETL (extract, transform, and load) functions.

The traditional ETL process extracts data from multiple sources, joins it with other relevant data, transforms it for analytical needs, and loads it into a data warehouse for subsequent analysis. Many organizations, including Intel, use a third-party ETL tool to perform this process. Rising costs associated with moving data and the increased dataset sizes prompted us to evaluate whether we could increase performance and achieve cost benefits by replacing our third-party ETL tool with our implementation of Hadoop, the Intel® Data Platform.

We first studied industry sources to learn the advantages and disadvantages of using Hadoop for big data ETL functions. We then tested with a real business use case that involved analyzing system logs. We compared costs and functional strengths of Hadoop and our third-party ETL tool.

We determined that using Hadoop for ETL functions works well for datasets that are coming from, passing through, or resting in Hadoop. Specifically, Hadoop makes sense for simple extract and load operations performed on those datasets. For non-Hadoop data, we do not recommend using Hadoop for ETL functions for these main reasons:

- Development, troubleshooting, and operational support for Hadoop-based features are still evolving and not as mature as our third-party ETL tools.
- Enterprise-grade features of Hadoop, specifically in the areas of performance, security, and quality of service (QoS), are not yet available.

Yatish Goel
BI Engineering Program Manager, Intel IT

Nghia Ngo
Big Data Capability Architect, Intel IT

Seshu Edala
Big Data Capability Engineer, Intel IT

Contents

Acronyms

ELT	extract, load, and transform
ETL	extract, transform, and load
HDFS	Hadoop Distributed File System
QoS	quality of service
RDBMS	relational database management system

Background

Intel extracts business value from big data, turning insights gained into competitive advantage. Part of this challenge involves mining big data from multiple sources, then cleansing, formatting, and loading it into a data warehouse for analysis, a process known as ETL (extract, transform, and load). To achieve this, Intel uses a third-party ETL solution.

Prompted by the following factors, we are now exploring ways to make ETL operations more cost effective:

- Growth in the volume, velocity, variability, and variety of the data
- Increase in costs associated with moving data
- New value from integrating structured and non-structured data
- Presence of an existing big data infrastructure that can already transform volumes of data

In addition to these factors, there is a shift away from traditional ETL to ELT (extract, load, and transform). This shift is mainly driven by big data, which follows the “store first, analyze later” model that is becoming the new standard. As shown in Figure 1, Hadoop performs the joins and transformation at the end of the process, so the order becomes ELT (extract, load, and transform). The ELT process brings the data into the storage container first and then identifies ways to get value from it. This is an important change that can be necessary to handle the large, fast, unstructured datasets coming in, whereas the traditional ETL process can create a bottleneck.

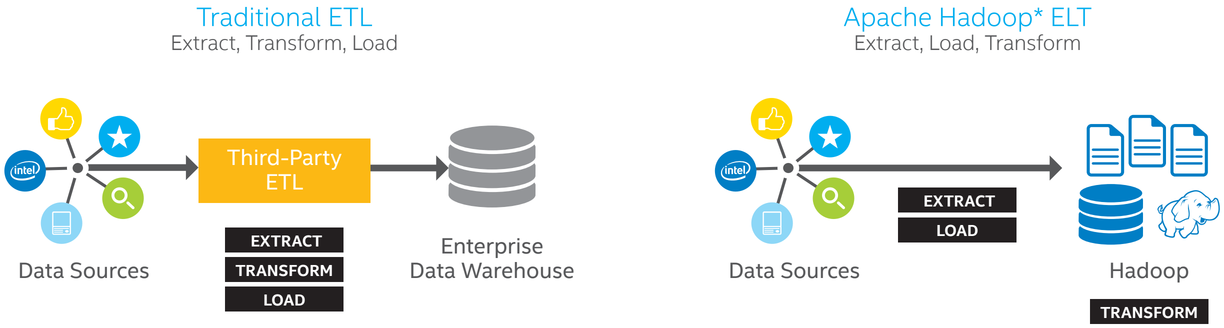

Figure 1. Intel IT evaluated Apache Hadoop* software as an option for performing traditional ETL (extract, transform, and load) functions. Using Hadoop, ETL becomes ELT (extract, load, and transform), with Hadoop processing and transforming the data at the end of the process.

To achieve the ELT process, we implemented the Intel® Data Platform, built on the Apache Hadoop* software, which is an integral part of our big data operations. As shown in Figure 2, the ELT process extracts data from multiple sources and loads it into Hadoop, where the transformation and processing takes place. Moving the transformation to the end of the process can eliminate the need for a separate ETL tool.

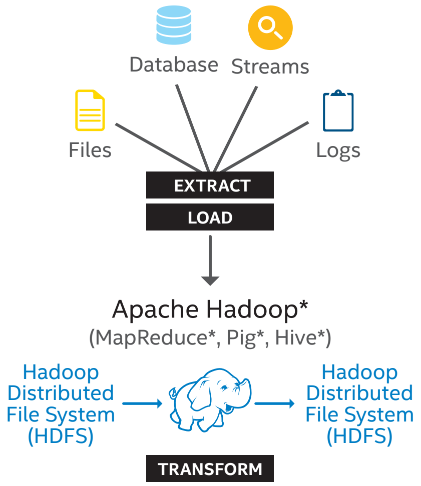
Database
Streams
Files
Logs
EXTRACT
LOAD
Apache Hadoop*
(MapReduce*, Pig*, Hive*)
Hadoop Distributed File System (HDFS)
Hadoop Distributed File System (HDFS)
TRANSFORM

Figure 2. With Apache Hadoop* software, the ETL (extract, transform, and load) process becomes ELT (extract, load, and transform). The ELT process extracts data from multiple sources and loads it into Hadoop, which then transforms and processes it.

Evaluation of Hadoop for ETL

In the first quarter of 2014, we briefly evaluated the following:

- We studied industry publications and case studies to evaluate Hadoop for ETL, including its features, capabilities, and limitations.
- We applied our findings to an internal use case, analyzing the functionality, costs, development effort, and future requirements as compared with our third-party ETL tool.
- We made recommendations and suggested next steps to management.

We explored whether we could derive performance and cost benefits by replacing our current third-party ETL tool with Hadoop.

Industry Analysis

After studying industry publications from Gartner, Forrester, and others, we identified several advantages and disadvantages to using Hadoop for ETL. Table 1 summarizes our findings.

Table 1. Advantages and Disadvantages of Apache Hadoop* Software for ETL (Extract, Transform, and Load) Functions

Advantages	Disadvantages
• Provides fast, low-cost processing for data coming from, passing through, or resting in Hadoop • Can handle structured and unstructured data • Is already present in many organization to meet other big data needs • Is well-supported in the open source developer community	• Is not able to perform transformations while data is in flight • Does not provide a user-friendly GUI development environment • Requires more code, which translates to more time to develop, support, and troubleshoot • Does not provide enterprise-level quality of service (QoS)

Case Study

To test the findings from our evaluation, we compared the performance of Hadoop for ETL to our third-party ETL tool. We used a real-world Intel use case that involved gathering and analyzing system logs and compared each area of functionality: extracting, loading, and transforming. We also studied the overall development environment and developed a cost comparison between Hadoop and our third-party ETL tool.

Functional Comparison

Table 2 shows a partial listing of the results, highlighting the areas where the supports levels differ between our third-party ETL tool and Hadoop for ETL.

In summary, we learned that Hadoop's extract and load functionalities are evolving and not as mature as our third-party ETL tool. In particular, using database APIs for extract and loads with Hadoop does not support operations on large-scale volumes.

Hadoop, when used with Apache Pig*, supports sufficient transformation capabilities, such as record reformatting, sorting, grouping, joining, filtering, merging, splitting, and combining record sets. Optionally, developers can also use Apache Hive*, a SQL-like language, to query and transform datasets, although Hive is used predominantly in data warehousing environments rather than ETL environments. Hive functionally supports similar transformation capabilities as Pig. However, Hadoop transformations using Pig or Hive require additional coding and developer skills to develop a framework upfront. Moreover, complex transformations require developing and integrating with user-defined functions that are not necessarily built into Hadoop.

Development Environment

Unlike our third-party ETL tool, Hadoop does not have a built-in GUI and is not a collaborative environment. To develop big data ETL, developers must use scripts to create the code, which increases the time it takes to develop, support, and troubleshoot the code. The Hadoop MapReduce* code requires approximately two to five times as many lines as SQL. Although there are open source GUI options for integration, enterprise support for such tools would require integration and additional investment. There are commercial tools available, but buying and incorporating another tool in place of our third-party tool is not cost effective.

Overall, operational support for Hadoop is not as mature as it is for our third-party ETL tool. For example, developers must use system tools for runtime monitoring and must manually monitor jobs and alerts; there are no user-friendly tools available. Debugging is not intuitive, and troubleshooting and fixing bugs require expertise. The restart and recovery processes are cumbersome, requiring developers to create logic within their code to handle both of these processes.

Table 2. Comparison of Support Levels for Third-Party ETL (Extract, Transform, and Load) Tool and Apache Hadoop* for ETL Functionality

● full support ◓ enhanced support ◔ limited support ○ no support

Functionality	Third-Party ETL Tool	Hadoop* for ETL
EXTRACT		
Extract from relational database management system (RDBMS)	●	◓
Extract from Hadoop	●	●
Hadoop Distributed File System (HDFS) to message service	◓	○
HDFS to XML	●	○
HDFS to web services	●	○
LOAD		
Load into RDBMS		
Full load	●	●
Delta load	●	◓
Load into Hadoop or files		
Full load	●	●
Delta load	◔	◔
TRANSFORM		
Complex type support	●	◓
Simple row projections		
Bulk data	●	●
Real-time data	●	◓
Aggregate operations	●	●
User-defined functions		
Row transformations	●	●
Sub-table aggregations	●	●
Windowing functions	◓	◔
Workflow control		
Triggers/conditional execution	●	◓
Pause/resume	●	◔
Incremental recovery/restore	●	◔
Advanced analytical functions (out-of-the-box string, crypto, date, and geo functions)	●	◓
Data quality and validation	●	◔

We also discovered that Hadoop does not provide an enterprise-level QoS. Performance, security, auditing, concurrency (write and read), and SQL compliance are still maturing in Hadoop and do not meet enterprise standards.

Cost Comparison

After analyzing the functionality, we compared the initial costs of the two tools, including the operational costs for the first year and for subsequent years (see Figure 3). We discovered that the costs are roughly the same for the initial implementation and first year of operation for the two tools, with Hadoop costing around two percent more than the third-party ETL tool. After the first year we predict that Hadoop will cost around 40 percent less to operate than the third-party ETL tool.

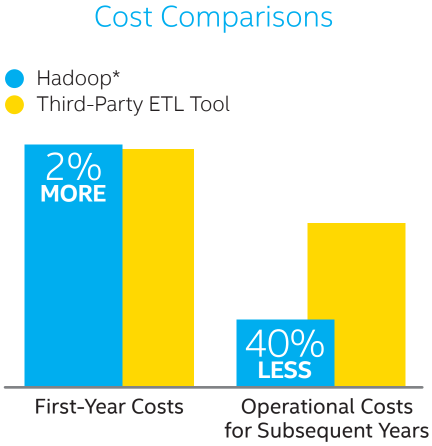

Figure 3. Our cost comparison revealed that costs for implementing Apache Hadoop* and the third-party ETL tool are roughly the same in the first year; however, in subsequent years it is predicted that Hadoop will cost about 40 percent less to operate.

Recommendations

After completing our industry analysis and exploring with a real use case, we recommend performing ETL functions with Hadoop when data comes from, passes through, or rests in Hadoop. For example, Intel IT and the Corporate Marketing Group are using Hadoop to analyze marketing campaign data to gain greater customer insight.[1] Additionally, we recommend using Hadoop for simple, lower-cost extract and load operations, where data is moving from one point to another and does not need to be joined and transformed in flight or be integrated in near-real time.

Working with data resting in Hadoop (in the Hadoop Distributed File System [HDFS]) is critical when selecting Hadoop as the ETL platform; when data resides elsewhere, we are deviating from the fundamental Hadoop philosophy of taking compute to the data. Traditional programming moves data to the processor, but the new functional programming provided by Hadoop MapReduce leaves data distributed, enabling massively parallel processing capabilities. If data does not rest in Hadoop, compute is separated from storage, which prevents you from taking advantage of the processing benefits that Hadoop MapReduce provides.

~40% LESS in subsequent years

After the first year we predict that Hadoop will cost less to operate than a third-party ETL tool.

For all other ETL functions, we recommend evaluating Hadoop case by case. We do not recommend using Hadoop for cases that require one or more of the following:

- Tier 1 and enterprise QoS
- Very high performance
- High and complex security
- Data quality monitors and profiling
- Absolute cleansing
- Data lineage and impact analysis
- Complex troubleshooting and debugging

[1] See the Intel IT white paper "Maximizing Marketing Insight through Big Data Analytics"

Next Steps

Next, we will present our recommendations to various stakeholders at Intel. Additionally, Intel IT's Business Intelligence Engineering guidelines have been updated to include Hadoop as an option for new extract and load projects.

We are completing the conversion of the system log use case to Hadoop by the end of the third quarter in 2014. After the conversion is done, we will refine our technical findings and cost analysis with the complete results.

For this study, we evaluated Hadoop 1.0. However, Hadoop is rapidly evolving in these areas even as it improves its enterprise acceptance rating:

- Operationally, Hadoop is improving by incorporating the Apache Hadoop YARN* resource negotiator. YARN delivers better compute and capacity management as well as co-tenancy of multiple processing frameworks like Apache Storm* and Apache Spark* on the same cluster.
- In addition to disk-backed MapReduce computing, other computational models are being added to Hadoop, such as in-memory mutable computing, directed acyclic graph architectures, and columnar storages. These models integrate batch analytics with real-time and interactive analytics.
- Hadoop is making it easier to integrate self-service advanced analytical capabilities such as text processing, machine learning, and data processing.

With these enhancements, Hadoop can help provide highly concurrent, near-real-time, low-latency data processing. We will continue to monitor big data Hadoop trends in the industry and reevaluate our ETL on Hadoop strategy in the future.

We will continue to monitor big data Hadoop trends in the industry and reevaluate our ETL on Hadoop strategy in the future.

Conclusion

Based on our evaluation of using Hadoop for enterprise ETL functions, we have learned that when data is local to Hadoop, we derive the most benefit in costs, functionality, and performance. We will continue to use Hadoop for ETL for datasets that are coming from, passing through, or resting in Hadoop. We also recommend using Hadoop for simple extract and load operations, and for operations that do not require transformations for data in flight.

For data that is not local to Hadoop, the benefits in functionality and performance diminish. Hadoop is still maturing and evolving in the areas of development environment, operational supportability, security, QoS, and SQL compliance. At this time, we intend to continue using both the Intel Data Platform and our third-party ETL tool to support various use cases.

For more information on Intel IT best practices, visit **www.intel.com/IT**.

IT@Intel

We connect IT professionals with their IT peers inside Intel. Our IT department solves some of today's most demanding and complex technology issues, and we want to share these lessons directly with our fellow IT professionals in an open peer-to-peer forum.

Our goal is simple: improve efficiency throughout the organization and enhance the business value of IT investments.

Follow us and join the conversation:

- Twitter
- #IntelIT
- LinkedIn
- IT Center Community

Visit us today at **intel.com/IT** or contact your local Intel representative if you would like to learn more.

Related Content

Visit **intel.com/IT** to find content on related topics:

- Extract, Transform, and Load Big Data with Apache Hadoop* paper
- Maximizing Marketing Insight through Big Data Analytics paper

THE INFORMATION PROVIDED IN THIS PAPER IS INTENDED TO BE GENERAL IN NATURE AND IS NOT SPECIFIC GUIDANCE. RECOMMENDATIONS (INCLUDING POTENTIAL COST SAVINGS) ARE BASED UPON INTEL'S EXPERIENCE AND ARE ESTIMATES ONLY. INTEL DOES NOT GUARANTEE OR WARRANT OTHERS WILL OBTAIN SIMILAR RESULTS.

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

0814/ACHA/KC/PDF

intel Look Inside.™